---((الفصل الخامس))---

في هذا الفصل، سنلقي نظرة على طرق مختلفة يمكنك من خلالها تكوين Kali Linux.

أولاً، في القسم 1.5، "تكوين الشبكة"، سنوضح لك كيفية تكوين إعدادات الشبكة باستخدام البيئة الرسومية وبيئة سطر الأوامر.

في القسم 2.5 "إدارة مستخدمي Unix ومجموعات Unix"، سنتكلم عن المستخدمين والمجموعات، ونوضح لك كيفية إنشاء وتعديل حسابات المستخدمين، وتعيين كلمات المرور، وتعطيل الحسابات، وإدارة المجموعات.

أخيرًا، سنناقش الخدمات في القسم 3.5، "تكوين الخوادم" وسنشرح كيفية إعداد الخدمات العامة والمحافظة عليها، ونركز أيضًا على الثلاث الخدمات المهمة للغاية وهي : SSH و PostgreSQL و Apache.

الإختصارا الواردة في هذا الفصل:

(SSH: **S**ecure **Sh**ell**)**, **(Gnome**: **G**NU **N**etwork **O**bject **M**odel **E**nvironment**)**

(GNU: **G**nu **N**ot **U**nix**)**, **(**Unix: ليس اختصارا**)**, **(IP**: **I**nternet **P**rotocol**)**

(Dhcp: **D**ynamic **H**ost **C**onfiguration **P**rotocol**)**

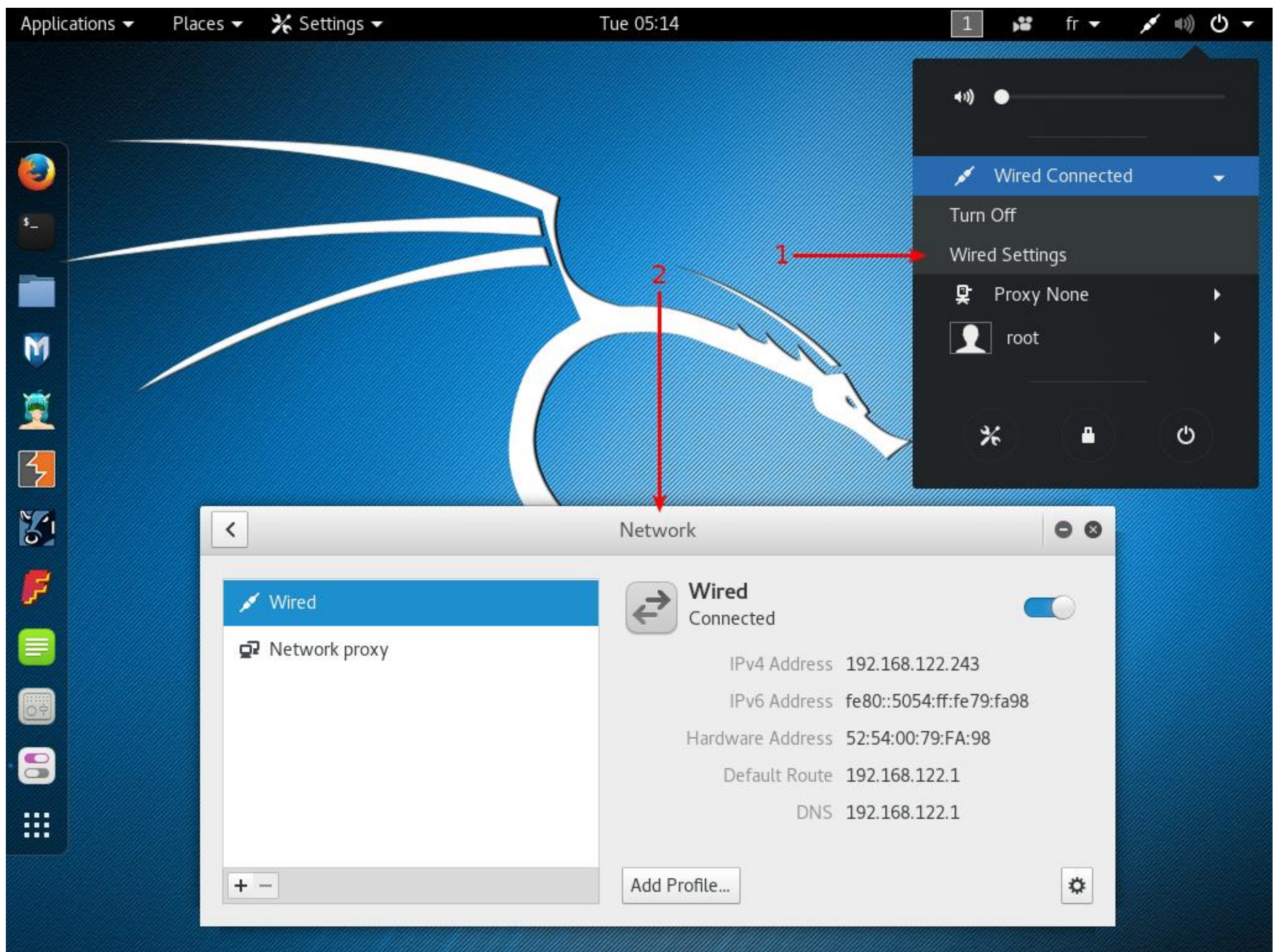
(MAC: **M**edia **A**ccess **C**ontrol**)**, **(SSID: S**ervice **S**et **ID**entifier**)**

(DNS: **D**omain **N**ame **S**ystem**)**, **(PPTP: P**oint-to-**P**oint **T**unneling **P**rotocol**)**.

1.5. تكوين الشبكة

١.١.٥. على سطح المكتب مع NetworkManager

في التثبيت العادي لسطح المكتب، سيكون لديك مدير شبكة –NetworkManager- مثبتاً بالفعل ويمكن التحكم فيه وتكوينه من خلال مركز التحكم في GNOME ومن خلال القائمة العلوية اليمنى كما هو موضح في الشكل 1.5. "شاشة تكوين الشبكة".



في شكل 1.5. "شاشة تكوين الشبكة".

يعتمد التكوين الافتراضي للشبكة على DHCP للحصول على عنوان IP وخادم DNS والبوابة، ولكن يمكنك استخدام رمز الترس في الزاوية اليمنى السفلية لتغيير التكوين بعدة طرق (على سبيل المثال: تعيين عنوان MAC والتبديل إلى إعداد static، قم بتمكين أو تعطيل IPv6، وإضافة موجهات "رواتر"). يمكنك إنشاء ملفات تعريف لحفظ تكوينات شبكة سلكية متعددة والتبديل بينها بسهولة. بالنسبة للشبكات اللاسلكية، ترتبط إعداداتها تلقائياً بمعرفها العام (SSID).

يعالج NetworkManager أيضًا الاتصالات عن طريق (الشبكة اللاسلكية واسعة النطاق "Wireless Wide Area Network" WWAN) وعن طريق أجهزة المودم باستخدام بروتوكول نقطة إلى نقطة عبر إيثرنت (PPPOE). أخيرًا وليس آخرًا، يوفر التكامل مع العديد من أنواع الشبكات الخاصة الافتراضية (VPN) من خلال المكونات الإضافية المخصصة: SSH و OpenVPN و Cisco's VPNC و PPTP و Strongswan.

حزم *-network-manager، معظمها غير مثبتة افتراضياً.

لاحظ أنك تحتاج إلى الحزم الملحقة بـ gnome- لتتمكن من تكوينها من خلال واجهة المستخدم الرسومية.

2.1.5. بسطر الأوامر باستخدام حزم ifupdown

بدلاً من ذلك، عندما تفضل عدم استخدام (أو لا يمكنك الوصول إلى) سطح مكتب رسومي، يمكنك تكوين الشبكة عن طريق حزمة ifupdown المثبتة بالفعل، والتي تتضمن أدوات **ifup** و**ifdown**. تقوم هذه الأدوات بقراءة التعريفات من ملف التكوين **/etc/network/interfaces** والتي هي في صميم البرنامج النصي **/etc/init.d/networking** الذي يقوم بتكوين الشبكة في وقت الإقلاع.

يمكن إلغاء تكوين كل أجهزة الشبكة التي تتم إدارته بواسطة ifupdown في أي وقت باستخدام **ifdown** *network-device*. يمكنك بعد ذلك تعديل **/etc/network/interfaces** وإعادة عمل الشبكة احتياطيًا (مع التكوين الجديد) باستخدام **ifup** *network-device*.

دعنا نلقي نظرة على ما يمكننا وضعه في ملف تكوين ifupdown. هناك توجيهان رئيسيان: **auto** *network-device*: الذي يخبر عن إعادة تهيئة لتكوين واجهة الشبكة تلقائيًا بمجرد توفرها. **iface** *network-device inet/inet6 type*: لتكوين واجهة معينة. على سبيل المثال، يبدو تكوين DHCP العادي كما يلي:

```
auto lo

iface lo inet loopback


auto eth0

iface eth0 inet dhcp
```

لاحظ أن التكوين الخاص لجهاز الاسترجاع (loopback) يجب أن يكون موجودًا دائمًا في هذا الملف. لتكوين عنوان IP ثابت، يجب عليك تقديم المزيد من التفاصيل مثل عنوان IP والشبكة وعنوان IP الخاص بالبوابة:

```
auto eth0

iface eth0 inet static

address 192.168.0.3

netmask 255.255.255.0

broadcast 192.168.0.255

network 192.168.0.0

gateway 192.168.0.1
```

بالنسبة للواجهات اللاسلكية، يجب أن يكون لديك حزمة wpasupplicant (مضمنة في Kali افتراضيًا)، والتي توفر العديد من خيارات **-*wpa** التي يمكن استخدامها في **/etc/network/interfaces.** ألق نظرة على /usr/share/doc/wpasupplicant/README.Debian.gz للحصول على أمثلة وشروحات. أكثر الخيارات شيوعًا هي wpa-ssid (الذي يحدد اسم الشبكة اللاسلكية للانضمام) وwpa-psk (الذي يحدد عبارة المرور أو المفتاح الذي يحمي الشبكة).

```
iface wlan0 inet dhcp

wpa-ssid MyNetWork

wpa-psk plaintextsecret
```

3.1.5. على سطر الأوامر باستخدام *systemd-networkd*

على الرغم من أن *ifupdown* هي الأداة التاريخية لدبيان، ولكنها لا تزال هي الأداة الافتراضية للخادم أو أي تثبيتات أخرى بسيطة، إلا أن هناك أداة أحدث تستحق التجربة وهي : *systemd-networkd*. ودمجها مع نظام *systemd* init يجعلها خيارًا جذابًا للغاية. لا يقتصر الأمر على التوزيعات المستندة على دبيان (على عكس *ifupdown*) وقد تم تصميمه ليكون صغيرًا جدًا وفعالًا وسهل التكوين نسبيًا إذا فهمت بنية ملفات وحدة النظام. يعد هذا خيارًا جذابًا بشكل خاص إذا كنت تعتقد أنه يصعب تهيئة *NetworkManager*.

يمكنك تكوين **systemd-networkd** عن طريق وضع ملفات **network**. في المجلد **/etc/systemd/network/**. بدلاً من ذلك، يمكنك استخدام **/lib/systemd/network/** لملفات الحزم أو **/run/systemd/network/** للملفات التي تم إنشاؤها في وقت التشغيل. تم توثيق تنسيق هذه الملفات في systemd.network(5). يشير قسم **Match** إلى واجهات الشبكة التي ينطبق عليها التكوين. يمكنك تحديد الواجهة بعدة طرق، بما في ذلك عن طريق عنوان التحكم في الوصول إلى الوسائط (MAC) أو نوع الجهاز. يحدد قسم **Network** تكوين الشبكة.

مثال 1.5 التكوين الثابت في **/etc/systemd/network/50-static.network**

[Match]

Name=enp2s0

[Network]

Address=192.168.0.15/24

Gateway=192.168.0.1

DNS=8.8.8.8

مثال 2.5 التكوين المستند على DHCP في **/etc/systemd/network/80-dhcp.network**

[Match]

Name=en*

[Network]

DHCP=yes

لاحظ أنه تم تعطيل **system-networkd** بشكل افتراضي، لذلك إذا كنت ترغب في استخدامه، يجب عليك تمكينه. يعتمد أيضًا على **systemd-resolved** من أجل التكامل الصحيح لدقة DNS، الأمر الذي يتطلب منك استبدال ملف **/etc/resolv.conf** بوصله رمزية لـ **/run/system/resolve/resolv.conf**، والذي تتم إدارته بواسطة **systemd-resolved**.

```
systemctl enable systemd-networkd
systemctl enable systemd-resolved
systemctl start systemd-networkd
systemctl start systemd-resolved
ln -sf /run/system/resolve/resolv.conf /etc/resolv.conf
```

على الرغم من أن **systemd-networkd** يعاني من بعض القيود، مثل عدم وجود دعم متكامل للشبكات اللاسلكية، يمكنك الاعتماد على تكوين **wpa_supplicant** خارجي موجود مسبقًا للدعم اللاسلكي. ومع ذلك، فهي مفيدة بشكل خاص في الـ containers والآلات الافتراضية "virtual machines"والتي تم تطويرها في الأصل للبيئات التي تعتمد فيها تهيئة شبكة الـ containers على تكوين شبكة مضيفها. في هذا السيناريو، يسهل **systemd-networkd** إدارة كلا الجانبين بطريقة منسقة مع الاستمرار في دعم جميع أنواع أجهزة الشبكة الافتراضية التي قد تحتاجها في هذا النوع من السيناريو انظر (5)systemd.netdev.

2.5. إدارة مستخدمي Unix ومجموعات Unix

تتكون قاعدة بيانات مستخدمي ومجموعات يونكس من ملفات نصية **/etc/passwd** (قائمة المستخدمين)، **/etc/shadow** (كلمات مرور مشفرة للمستخدمين)، **/etc/group** (قائمة المجموعات)، و **/etc/gshadow** (كلمات مرور مشفرة للمجموعات). تنسيقاتها موثقة في passwd (5) و shadow (5) و group (5) و gshadow (5) على التوالي. بينما يمكن تحرير هذه الملفات يدوياً باستخدام أدوات مثل **`vipw`** و **`vigr`**، هناك أدوات ذات مستوى أعلى لإجراء العمليات الأكثر شيوعاً.

استخدام getent لاستشارة قاعدة بيانات المستخدم

يتحقق الأمر **`getent`** (get entries) من قواعد بيانات النظام (بما في ذلك قواعد البيانات الخاصة بالمستخدمين والمجموعات) باستخدام وظائف المكتبة المناسبة، والتي بدورها تستدعي وحدات خدمة تبديل الاسم (NSS) المكونة في الملف /etc/nsswitch.conf. يأخذ الأمر مدخل أو اثنتين: اسم قاعدة البيانات للتحقق، ومفتاح بحث محتمل. وبالتالي، فإن الأمر **`getent passwd kaliuser1`** سيعيد المعلومات من قاعدة بيانات المستخدم المتعلقة بالمستخدم **kaliuser1**.

```
root@kali:~# getent passwd kaliuser1
kaliuser1:x:1001:1001:Kali User,4444,123-867-5309,321-867-5309:/home/kaliuser1:/bin/bash
```

١.٢.٥. إنشاء حسابات المستخدمين

قد تحتاج أحيانا إلى إنشاء حسابات مستخدم غير مميزة لأسباب مختلفة، خاصة إذا كنت تستخدم Kali كنظام تشغيل أساسي. الطريقة الأكثر شيوعًا لإضافة مستخدم هي الأمر **adduser**، الذي يطلب مدخل مطلوب وهو: اسم المستخدم، للمستخدم الجديد الذي ترغب في إنشائه.

يطرح أمر **adduser** بعض الأسئلة قبل إنشاء الحساب ولكن استخدامه واضح إلى حد ما. يتضمن ملف التكوين الخاص به، **/etc/adduser.conf**، العديد من الإعدادات المثيرة للاهتمام. يمكنك، على سبيل المثال، تحديد نطاق معرفات المستخدم (UIDs) التي يمكن استخدامها، وإملاء ما إذا كان المستخدمون يشتركون في مجموعة مشتركة أم لا، وتحديد الصدفات الافتراضية، والمزيد.

يؤدي إنشاء حساب إلى تشغيل محتوى المجلد الرئيسي للمستخدم بمحتويات النموذج **/etc/skel/**. يوفر ذلك للمستخدم مجموعة من المجلدات القياسية وملفات التكوين.

في بعض الحالات، سيكون من المفيد إضافة مستخدم إلى مجموعة (بخلاف المجموعة الرئيسية الافتراضية) لمنح أذونات إضافية. على سبيل المثال، المستخدم الذي تم تضمينه في مجموعة *sudo* لديه امتيازات إدارية كاملة من خلال الأمر **sudo**. يمكن تحقيق ذلك باستخدام أمر مثل: **adduser** *user group*.

٢.٢.٥. تعديل حساب موجود أو كلمة مرور

تسمح الأوامر التالية بتعديل المعلومات المخزنة في حقول محددة من قواعد بيانات المستخدم:

passwd - يسمح للمستخدم العادي بتغيير كلمة المرور الخاصة به، والتي بدورها تقوم بتحديث ملف **/etc/shadow**. || أو كلمة مرور مستخدم أخر، مثلا: sudo passwd al3mamy||

chfn– (تغيير الاسم الكامل)، المحجوز للمستخدم الفائق (الجذر)، يعدل حقل **GECOS**، أو حقل "معلومات عامة".

chsh – (تغيير الصدفة) يغير واجهة تسجيل دخول المستخدم. ومع ذلك، ستقتصر الخيارات المتاحة على تلك المدرجة في **/etc/shells**؛ المسؤول، من ناحية أخرى، غير ملزم بهذا التقييد ويمكنه تعيين shell لأي برنامج تم اختياره.

chage – (تغيير العمر) يسمح للمسؤول بتغيير إعدادات انتهاء صلاحية كلمة المرور بتمرير اسم المستخدم كمدخل أو سرد الإعدادات الحالية باستخدام الخيار **-l** user. بدلاً من ذلك، يمكنك أيضًا فرض انتهاء صلاحية كلمة المرور باستخدام الأمر **passwd -e** user، مما يجبر المستخدم على تغيير كلمة المرور الخاصة به في المرة التالية التي يقوم فيها بتسجيل الدخول.

٣.٢.٥. تعطيل حساب

قد تجد نفسك بحاجة إلى تعطيل حساب (حظر مستخدم) كإجراء تأديبي، لأغراض التحقيق، أو ببساطة في حالة الغياب المطول أو النهائي للمستخدم. يعني الحساب المعطل أن المستخدم لا يمكنه تسجيل الدخول أو الوصول إلى الجهاز. يظل الحساب كما هو على الجهاز ولا يتم حذف أي ملفات أو بيانات، ببساطة لا يمكن الوصول إليها. يتم تحقيق ذلك باستخدام الأمر **passwd -l** *user* (القفل "lock"). تتم إعادة تمكين الحساب بطريقة مماثلة، مع الخيار **u-** (إلغاء القفل).

--||--

لقفل الحساب: `passwd -l user`

لإلغاء قفل الحساب: `passwd --unlock user`

لحذف كلمة المرور: `passwd --delete user`

وغيره .. تحقق من `man passwd` وانظر للخيارات المتاحة.

--||--

٤.٢.٥. إدارة مجموعات يونكس

يضيف الأمر **addgroup** و **delgroup** يحذف مجموعة، على التوالي. يعدل الأمر **groupmod** معلومات المجموعة (رقم تعريفها -**gid**- أو معرفها). يقوم الأمر **gpasswd** *group* بتغيير كلمة مرور المجموعة، بينما يقوم الأمر **gpasswd -r** *group* بحذفها.

العمل على عدة مجموعات

قد يكون كل مستخدم عضوا في العديد من المجموعات. يتم إنشاء المجموعة الرئيسية للمستخدم بشكل افتراضي أثناء التكوين الأولي للمستخدم. بشكل افتراضي، ينتمي كل ملف ينشئه المستخدم إلى المستخدم وكذلك إلى المجموعة الرئيسية للمستخدم. هذا ليس مرغوبًا دائمًا؛ على سبيل المثال، عندما يحتاج المستخدم للعمل في مجلد مشترك من قبل مجموعة غير مجموعته الرئيسية. في هذه الحالة، يحتاج المستخدم إلى تغيير المجموعات باستخدام أحد الأوامر التالية: **newgrp**، الذي يبدأ بصدفة جديدة، أو **sg**، والذي يقوم ببساطة بتنفيذ أمر باستخدام المجموعة البديلة المزودة. تسمح هذه الأوامر أيضًا للمستخدم بالانضمام إلى مجموعة لا ينتمي إليها حاليًا. إذا كانت المجموعة محمية بكلمة مرور، فسوف تحتاج إلى توفير كلمة المرور المناسبة قبل تنفيذ الأمر.

بدلاً من ذلك، يمكن للمستخدم تعيين بت **setgid** في المجلد، مما يؤدي إلى أن تنتمي الملفات التي تم إنشاؤها في هذا المجلد تلقائيًا إلى المجموعة الصحيحة.

يعرض الأمر **id** الحالة الحالية للمستخدم، مع معرفه الشخصي (متغير **uid**)، والمجموعة الرئيسية الحالية (متغير **gid**)، وقائمة المجموعات التي ينتمون إليها (متغير **groups**).

٣.٥. تكوين الخدمات

في هذا القسم، سنلقي نظرة على الخدمات (تسمى أحيانًا daemons)، أو البرامج التي تعمل كعمليات في الخلفية وتؤدي وظائف متنوعة للنظام. سنبدأ بمناقشة ملفات التكوين وسنشرع في شرح كيفية عمل بعض الخدمات المهمة (مثل SSH وPostgreSQL وApache) وكيف يمكن تكوينها.

١.٣.٥. تكوين برنامج معين

عندما تريد تكوين حزمة غير معروفة، يجب عليك متابعة هذه المراحل. أولاً، يجب عليك قراءة ما وثقه مشرف الحزمة. يعد ملف **/usr/share/doc/***package***/README.Debian** مكانًا جيدًا للبدء. غالبًا ما يحتوي هذا الملف على معلومات حول الحزم، بما في ذلك المؤشرات التي قد تحيلك إلى وثائق أخرى. غالبًا ما توفر لك الكثير من الوقت، وتجنب الكثير من الإحباط، من خلال قراءة هذا الملف أولاً لأنه غالبًا ما يوضح تفاصيل الأخطاء والحلول الأكثر شيوعًا لمعظم المشاكل الشائعة.

بعد ذلك، يجب عليك إلقاء نظرة على الوثائق الرسمية للبرنامج. راجع القسم ١.٦. "مصادر التوثيق" للحصول على نصائح حول كيفية العثور على مصادر توثيق مختلفة. يعطي الأمر:

dpkg -L *package*

قائمة بالملفات المضمنة في الحزمة؛ لذلك يمكنك تحديد الوثائق المتاحة بسرعة (بالإضافة إلى ملفات التكوين الموجودة في /etc/). أيضًا الأمر: **dpkg -s** *package* يعرض البيانات الوصفية للحزمة وتعرض أي حزم ممكنة موصى بها أو مقترحة؛ هناك، يمكنك العثور على وثائق أو أداة مساعدة من شأنها تسهيل تكوين البرنامج.

أخيرًا، غالبًا ما يتم توثيق ملفات التكوين ذاتيًا من خلال العديد من التعليقات التفسيرية التي توضح بالتفصيل مختلف القيم الممكنة لكل إعداد تكوين. في بعض الحالات، يمكنك الحصول على البرنامج وتشغيله عن طريق إلغاء تعليق سطر واحد فقط في ملف التكوين. في حالات أخرى، يتم توفير أمثلة لملفات التكوين في المجلد **/usr/share/doc/***package***/examples/**. قد تكون بمثابة أساس لملف التكوين الخاص بك.

٢.٣.٥. تكوين SSH لتسجيلات الدخول عن بعد

يسمح لك SSH بتسجيل الدخول إلى الجهاز عن بُعد أو نقل الملفات أو تنفيذ الأوامر. الأداة القياسية هي (ssh) وأما الخدمة (sshd) للاتصال بالأجهزة عن بعد.

أثناء تثبيت حزمة opensh-server افتراضيًا، يتم تعطيل خدمة SSH افتراضيًا وبالتالي لا يتم تشغيلها في وقت الإقلاع. يمكنك بدء تشغيل خدمة SSH يدويًا بكتابة الأمر:

```
systemctl start ssh
```

أو تهيئتها للبدء في وقت الإقلاع باستخدام الأمر:

```
systemctl enable ssh
```

خدمة SSH لها تكوين افتراضي معقول نسبيًا، ولكن نظرًا لقدراتها القوية وطبيعتها الحساسة، من الجيد معرفة ما يمكنك القيام به في ملف التكوين الخاص بها، **/etc/ssh/sshd_config**. تم توثيق جميع الخيارات في sshd_config (5).

يعطل التكوين الافتراضي عمليات تسجيل الدخول المستندة إلى كلمة المرور للمستخدم الجذر، مما يعني أنه يجب عليك أولاً إعداد مفاتيح SSH باستخدام **ssh-keygen**. يمكنك تمديد هذا إلى جميع المستخدمين عن طريق تعيين **PasswordAuthentication** إلى **no**، أو يمكنك رفع هذا القيد عن طريق تغيير **PermitRootLogin** إلى **yes** (بدلاً من كلمة المرور المحظورة الافتراضية). تستمع خدمة SSH بشكل افتراضي على المنفذ (port) 22 ولكن يمكنك تغيير ذلك باستخدام توجيه **Port**.

لتطبيق الإعدادات الجديدة، يجب كتابة الأمر **systemctl reload ssh**.

توليد مفاتيح مضيف SSH جديدة

يحتوي كل خادم SSH على مفاتيح التشفير الخاصة به؛ يتم تسميتها "مفاتيح مضيف SSH" ويتم تخزينها في **/etc/ssh/ssh_host_***. يجب الحفاظ على خصوصيتها إذا كنت تريد السرية ولا يجب مشاركتها مع أجهزة متعددة.

عندما تقوم بتثبيت النظام الخاص بك عن طريق نسخ صورة قرص كاملة (بدلاً من استخدام برنامج debian-installer)، فقد تحتوي الصورة على مفاتيح مضيف SSH تم إنشاؤها مسبقًا والتي يجب عليك استبدالها بمفاتيح تم إنشاؤها حديثًا. من المحتمل أن تأتي الصورة أيضًا بكلمة مرور جذر افتراضية تريد إعادة تعيينها في نفس الوقت. يمكنك القيام بكل ذلك باستخدام الأوامر التالية:

```
#passwd [...]
#rm /etc/ssh/ssh_host*_
#dpkg-reconfigure openssh-server
#service ssh restart
```

٣.٣.٥. تكوين قواعد بيانات PostgreSQL

PostgreSQL هو خادم قاعدة بيانات. نادرًا ما يكون مفيدًا من تلقاء نفسه ولكن يتم استخدامه من قبل العديد من الخدمات الأخرى لتخزين البيانات. ستصل هذه الخدمات بشكل عام إلى خادم قاعدة البيانات عبر الشبكة وتتطلب عادة بيانات اعتماد المصادقة لتكون قادرة على الاتصال. وبالتالي يتطلب إعداد هذه الخدمات إنشاء قواعد بيانات PostgreSQL وحسابات المستخدمين مع الامتيازات المناسبة لقاعدة البيانات. حتى نتمكن من القيام بذلك، نحتاج إلى تشغيل الخدمة، لذا دعنا نبدأ بالأمر:

```
systemctl start postgresql
```

دعم العديد من إصدارات PostgreSQL

تسمح حزمة PostgreSQL بتثبيت نسخ متعددة من خادم قاعدة البيانات. من الممكن أيضًا التعامل مع *clusters* متعددة (cluster هي مجموعة من قواعد البيانات التي يقدمها نفس مدير البريد "**postmaster**"). لتحقيق ذلك، يتم تخزين ملفات التكوين في **/etc/postgresql/***version*/*cluster-name*/.

من أجل تشغيل الـ clusters جنبًا إلى جنب، يتم تعيين رقم المنفذ التالي المتاح لكل مجموعة جديدة (عادةً ٥٤٣٣ للمجموعة الثانية). ملف **postgresql.service** عبارة عن صدفة فارغة، مما يجعل من السهل العمل على كل المجموعات "clusters" معًا حيث أن لكل مجموعة وحدتها الخاصة (**postgresql@***version*-*cluster*.**service**).

١.٣.٣.٥. نوع الاتصال ومصادقة العميل

بشكل افتراضي، يستمع PostgreSQL للاتصالات الواردة بطريقتين: على منفذ TCP 5432 لواجهة المضيف المحلي وعلى المقبس " socket" المستند للملفات **/var/run/postgresql/.s.PGSQL.5432**. يمكن تكوين ذلك في **postgresql.conf** مع توجيهات متنوعة: **listen_addresses** للعنوان الذي تريد الاستماع منه، **port** لمنفذ TCP، و **unix_socket_directories** لتعريف المجلد حيث يتم إنشاء المقابس المستندة إلى الملفات.

اعتمادًا على كيفية الاتصال، يتم مصادقة العملاء بطرق مختلفة. يحدد ملف التكوين **pg_hba.conf** من الذي يسمح له بالاتصال على كل مقبس وكيفية مصادقته. بشكل افتراضي، تستخدم الاتصالات على مأخذ التوصيل المستند إلى الملفات حساب مستخدم Unix كاسم مستخدم PosgreSQL، ويفترض أنه لا توجد مصادقة أخرى مطلوبة. في اتصال TCP، يطلب PostgreSQL من المستخدم المصادقة باستخدام اسم مستخدم وكلمة مرور (على الرغم من أنه ليس اسم مستخدم/كلمة مرور Unix ولكن بدلاً من ذلك واحد يديره PostgreSQL نفسه).

مستخدم **postgres** خاص ولديه امتيازات إدارية كاملة على جميع قواعد البيانات. سنستخدم هذه الهوية لإنشاء مستخدمين جدد وقواعد بيانات جديدة.

٢.٣.٣.٥. إنشاء المستخدمين وقواعد البيانات

يضيف الأمر **createuser** مستخدمًا جديدًا ويزيل **dropuser** المستخدم. وبالمثل، يضيف الأمر **createdb** قاعدة بيانات جديدة ويزيل **dropdb** قاعدة بيانات. كل من هذه الأوامر لها صفحات يدوية خاصة بها ولكننا سنناقش بعض الخيارات هنا. يعمل كل أمر على الكتلة " cluster" الافتراضية (يعمل على المنفذ 5432) ولكن يمكنك تمرير:

--port=*port*

لتعديل المستخدمين وقواعد البيانات الخاصة بالكتلة البديلة.

يجب أن تتصل هذه الأوامر بخادم PostgreSQL للقيام بعملهم ويجب أن تتم مصادقتهم كمستخدم يتمتع بامتيازات كافية ليتمكنوا من تنفيذ العملية المحددة. أسهل طريقة لتحقيق ذلك هي استخدام حساب Unix **postgres** والاتصال عبر المقبس المستند إلى الملفات " file-based socket":

```
# su - postgres
$ createuser -P king_phisher
Enter password for new role:
Enter it again:
$ createdb -T template0 -E UTF-8 -O king_phisher king_phisher
$ exit
```

في المثال السابق، يطلب الخيار **P-** من الأمر **createuser** تعيين كلمة مرور جديدة لحساب **king_phisher** بمجرد إنشائه. بالنظر إلى الأمر **createdb**، يحدد الخيار **O-** المستخدم الذي يمتلك قاعدة البيانات الجديدة (الذي يتمتع بالتالي بحقوق كاملة لإنشاء الجداول ومنح

الأذونات وما إلى ذلك). نريد أيضًا أن نكون قادرين على استخدام سلاسل Unicode، لذلك نضيف الخيار **E UTF-8-** لضبط الترميز، والذي بدوره يتطلب منا استخدام خيار **T-** لاختيار قالب قاعدة بيانات آخر.

يمكننا الآن اختبار إمكانية الاتصال بقاعدة البيانات عبر الاستماع إلى مأخذ التوصيل على المضيف المحلي (**-h localhost**) كمستخدم king_phisher (**-U king_phisher**):

```
# psql -h localhost -U king_phisher king_phisher
Password for user king_phisher:
psql (9.5.2)
SSL connection (protocol: TLSv1.2, cipher: ECDHE-RSA-AES256-GCM-SHA384, bits: 256, compression: off)
Type "help" for help. king_phisher=>
```

كما ترى، نجح الاتصال.

٥.٣.٣.٣. إدارة مجموعات PostgreSQL

أولاً، تجدر الإشارة إلى أن مفهوم "مجموعة -cluster– PostgreSQL" هو إضافة خاصة بـ Debian ولن تجد أي إشارة لهذا المصطلح في وثائق PostgreSQL الرسمية. من وجهة نظر أدوات PostgreSQL، فإن هذه المجموعة هي مجرد مثيل لخادم قاعدة بيانات يعمل على منفذ معين.

ومع ذلك، توفر حزمة ديبيان postgresql الشائعة أدوات متعددة لإدارة هذه المجموعات:

pg_createcluster, **pg_dropcluster**, **pg_ctlcluster**,

pg_upgradecluster, **pg_renamecluster**, **pg_lsclusters.**

لن نغطي جميع هذه الأدوات هنا، ولكن يمكنك الرجوع إلى الصفحات اليدوية الخاصة بها لمزيد من المعلومات.

ما يجب أن تعرفه هو أنه عندما يتم تثبيت إصدار رئيسي جديد من PostgreSQL على نظامك، فإنه سيقوم بإنشاء مجموعة جديدة تعمل على المنفذ التالي (عادة 5433) وستستمر في استخدام الإصدار القديم حتى تقوم بترحيل قواعد البيانات الخاصة بك من المجموعة القديمة إلى الجديدة.

يمكنك استرداد قائمة بجميع المجموعات وحالتها باستخدام أمر: **pg_lsclusters**. الأهم من ذلك، يمكنك أتمتة ترحيل نظامك إلى أحدث إصدار من PostgreSQL باستخدام:

```
pg_upgradecluster old-version cluster-name
```

لكي ينجح هذا، قد تحتاج أولاً إلى إزالة نظام المجموعة (empty) الذي تم إنشاؤه من أجل الإصدار الجديد (باستخدام: **pg_dropcluster** *new-version cluster-name*).
لا يتم إسقاط المجموعة القديمة في العملية، ولكن لن يتم بدء تشغيلها تلقائيًا أيضًا. يمكنك إسقاطها بمجرد التحقق من أن المجموعة التي تمت ترقيتها تعمل بشكل جيد.

٤.٣.٥. تكوين أباتشي

يشتمل التثبيت النموذجي لـ Kali Linux على خادم الويب Apache، الذي توفره حزمة apache2. كونها خدمة شبكة، يتم تعطيلها بشكل افتراضي. يمكنك تشغيله يدويًا باستخدام: **systemctl start apache2**.

مع انتشار الكثير والكثير من تطبيقات الويب صار من المهم أن يكون لديك بعض المعرفة بـ Apache من أجل استضافة هذه التطبيقات، سواء للاستخدام المحلي أو لإتاحتها عبر الشبكة.

Apache هو خادم وحدات -modular server- ويتم تنفيذ العديد من الميزات بواسطة وحدات -modules- خارجية يقوم البرنامج الرئيسي بتحميلها أثناء التهيئة. يتيح التكوين الافتراضي فقط الوحدات -modules- الأكثر شيوعًا، ولكن تمكين الوحدات الجديدة يتم بسهولة عن طريق تشغيل **a2enmod** *module*. استخدم **a2dismod** *module* لتعطيل الوحدة. تقوم هذه البرامج في الواقع بإنشاء (أو حذف) روابط رمزية فقط في:

||**a2enmod** "**a**pache **2** **en**able **mod**ule". **a2dismod** "**a**pache **2** **dis**able **mod**ule"||

/etc/apache2/mods-enabled/

مشيرة إلى الملفات الحقيقية (المخزنة في **/etc/apache2/mods-available/**).

هناك العديد من الوحدات المتاحة، ولكن هناك اثنتان تستحق النظر الأولي: PHP وSSL. يتم تنفيذ تطبيقات الويب المكتوبة باستخدام PHP بواسطة خادم الويب Apache بمساعدة الوحدة المخصصة التي توفرها حزمة libapache-mod-php، وعند تثبيتها تمكن الوحدة تلقائيا.

يتضمن Apache 2.4 وحدة SSL المطلوبة لـ HTTP الآمن (HTTPS) خارج الصندوق. يجب أولاً تمكينه باستخدام: **a2enmod ssl**، ثم يجب إضافة التوجيهات المطلوبة لملفات التكوين. يتوفر مثال التكوين في **/etc/apache2/sites-available/default-ssl.conf**. راجع http://httpd.apache.org/docs/2.4/mod/mod_ssl.html لمزيد من المعلومات.

يمكن العثور على القائمة الكاملة لوحدات Apache القياسية عبر الإنترنت في http://httpd.apache.org/docs/2.4/mod/index.html.

باستخدام التكوين الافتراضي، يستمع خادم الويب على المنفذ 80 (كما تم تكوينه في **/etc/apache2/ports.conf**)، وصفحات الخادم من المجلد **/var/www/html/** بشكل افتراضي (كما تم تكوينه في **/etc/apache2/sites-enabled/000-default.conf**).

١.٤.٣.٥. تكوين المضيفين الافتراضيين

المضيف الافتراضي هو هوية إضافية لخادم الويب. يمكن أن تخدم عملية أباتشي نفسها مواقع ويب متعددة (مثل www.kali.org و www.offensive-security.com) لأن طلبات HTTP تتضمن كلاً من اسم موقع الويب المطلوب وعنوان URL المحلي (تُعرف هذه الميزة باسم: *namebased virtual hosts*).

يتيح التكوين الافتراضي لـ Apache 2 تمكين name-based virtual hosts. بالإضافة إلى ذلك، يتم تعريف مضيف افتراضي افتراضيا في ملف **/etc/apache2/sites-enabled/000-default.conf**، سيتم استخدام هذا المضيف الافتراضي إذا لم يتم العثور على مضيف مطابق للطلب الذي أرسله العميل.

مهم

سيتم دائمًا تقديم الطلبات المتعلقة بالمضيفات الافتراضية غير المعروفة بواسطة المضيف الافتراضي المحدد أولا، ولهذا السبب تقوم الحزمة بشحن ملف تكوين **000-default.conf**، والذي يتم فرزه أولاً بين جميع الملفات الأخرى التي قد تقوم بإنشائها.

يتم بعد ذلك وصف كل مضيف افتراضي إضافي بواسطة ملف مخزن في **/etc/apache2/sites-available/**. عادةً ما تتم تسمية الملف باسم موقع الويب متبوعًا بامتداد **conf**. (على سبيل المثال: **www.example.com.conf**). يمكنك بعد ذلك تمكين المضيف الافتراضي الجديد باستخدام: *www.example.com* **a2ensite**. فيما يلي الحد الأدنى من تكوين المستضيف الافتراضي لموقع ويب يتم تخزين ملفاته في **/srv/www.example.com/www/** (محدد بخيار **DocumentRoot**):

```
ServerName www.example.com
ServerAlias example.com
DocumentRoot /srv/www.example.com/www
```

قد تفكر أيضًا في إضافة توجيهات `CustomLog` و `ErrorLog` لتكوين Apache لإخراج سجلات الدخول في ملفات مخصصة للمضيف الافتراضي.

٢.٤.٣.٥. توجيهات شائعة الإستخدام

يستعرض هذا القسم بعض توجيهات تكوين Apache شائعة الاستخدام.

عادة ما يتضمن ملف التكوين الرئيسي العديد من كتل **Directory** -مجلد-، أنها تسمح بتحديد سلوكيات مختلفة للخادم حسب موقع الملف الذي يتم تقديمه. تتضمن مثل هذه الكتلة الشائعة **Options** و **AllowOverride**:

```
Options Includes FollowSymLinks
AllowOverride All
DirectoryIndex index.php index.html index.htm
```

يحتوي التوجيه **DirectoryIndex** على قائمة بالملفات التي يجب تجربتها عندما يطابق طلب العميل مجلدا. يتم استخدام أول ملف موجود في القائمة وإرساله كرد.

ويتبع توجيه **Options** قائمة من الخيارات للتمكين. تقوم القيمة **None** بتعطيل جميع الخيارات، وبالمثل، فإن **All** يمكّنهم جميعًا باستثناء **MultiViews**. تشمل الخيارات المتاحة:

- **ExecCGI** - يشير إلى أنه يمكن تنفيذ البرامج النصية CGI.
- **FollowSymLinks** - تخبر الخادم أنه يمكنه اتباع الروابط الرمزية، وأن الاستجابة يجب أن تحتوي على محتويات هدف هذه الروابط.
- **SymLinksIfOwnerMatch** - يخبر الخادم أيضًا باتباع الروابط الرمزية، ولكن فقط عندما يكون للرابط وهدفه المالك نفسه.

- ❖ **Includes**- يمكّن تضمين جانب الخادم -*Server Side Includes*- (SSI). هذه توجيهات مضمنة في صفحات HTML وتنفيذها على الفور لكل طلب.
- ❖ **Indexes**– تطلب من الخادم إدراج محتويات المجلد إذا كان طلب HTTP الذي أرسله العميل يشير إلى مجلد بدون ملف فهرس (أي عندما لا توجد ملفات مذكورة في توجيه **DirectoryIndex** في هذا المجلد).
- ❖ **MultiViews** - تتيح التفاوض -negotiation- على المحتوى، يمكن استخدام هذا من قبل الخادم لإرجاع صفحة ويب مطابقة للغة المفضلة كما تم تكوينه في المستعرض.

٥.٣.٤.٢.١. طلب المصادقة

في بعض الحالات، يجب تقييد الوصول إلى جزء من موقع ويب، لذلك يتم منح حق الوصول إلى المحتويات للمستخدمين الشرعيين فقط الذين يقدمون اسم مستخدم وكلمة مرور.

يحتوي ملف **.htaccess** على توجيهات تكوين Apache التي يتم فرضها في كل مرة يتعلق فيها الطلب بعنصر من المجلد حيث يتم تخزين ملف **.htaccess**. هذه التوجيهات متكررة، مما يوسع النطاق ليشمل جميع المجلدات الفرعية.

معظم التوجيهات التي يمكن أن تحدث في كتلة **Directory** قانونية أيضًا في ملف **.htaccess**. يسرد الأمر **AllowOverride** جميع الخيارات التي يمكن تمكينها أو تعطيلها عن طريق **.htaccess**. الاستخدام الشائع لهذا الخيار هو تقييد **ExecCGI**، بحيث يختار المسؤول المستخدمين المسموح لهم بتشغيل البرامج تحت هوية خادم الويب (مستخدم الـ **www-data**).

مثال ٥.٣. **.htaccess** ملف يتطلب المصادقة

```
Require valid-user
AuthName "Private directory"
AuthType Basic
AuthUserFile /etc/apache2/authfiles/htpasswd-private
```

لا توفر المصادقة الأساسية -**Basic**- الأمان

يتمتع نظام المصادقة المستخدم في المثال السابق (**Basic**) بالحد الأدنى من الأمان حيث يتم إرسال كلمة المرور بنص واضح (يتم ترميزها فقط كـ *base64*، وهو ترميز بسيط بدلاً من أسلوب تشفير). وتجدر الإشارة أيضًا إلى أن المستندات التي تحميها هذه الآلية أيضًا تمر عبر الشبكة بشكل واضح. إذا كان الأمان مهمًا، فيجب تشفير جلسة HTTP بالكامل باستخدام طبقة النقل الآمنة (TLS).

يحتوي الملف **/etc/apache2/authfiles/htpasswd-private** على قائمة بالمستخدمين وكلمات المرور، يتم التلاعب بها عادة باستخدام الأمر **htpasswd**. على سبيل المثال، يتم استخدام الأمر التالي لإضافة مستخدم أو تغيير كلمة المرور الخاصة به:

```
# htpasswd /etc/apache2/authfiles/htpasswd-private user
New password: Re-type new password: Adding password for user user
```

٢.٢.٤.٣.٥. تقييد الوصول

يتحكم التوجيه **Require** في قيود الوصول إلى المجلد (والمجلدات الفرعية الخاصة به، بشكل متكرر).

يمكن استخدامه لتقييد الوصول على أساس العديد من المعايير، سنتوقف عند وصف تقييد الوصول استنادًا إلى عنوان IP للعميل ولكن يمكن جعله أكثر قوة من ذلك، خاصة عندما يتم دمج العديد من التوجيهات المطلوبة -**Require**- داخل كتلة **RequireAll**.

على سبيل المثال، يمكنك تقييد الوصول إلى الشبكة المحلية باستخدام التوجيه التالي:

```
Require ip 192.168.0.0/16
```

4.5. إدارة الخوادم

يستخدم كالي **systemd** كنظام خاص به، وهو ليس مسؤولًا فقط عن تسلسل الإقلاع، ولكنه يعمل أيضًا بشكل دائم كمدير خوادم كامل الميزات لبدء ومراقبة الخدمات.

يمكن الاستعلام عن **systemd** والتحكم فيه باستخدام **systemctl**. بدون أي مدخلات، يقوم بتشغيل الأمر **systemctl list-units** الذي ينتج قائمة *بالوحدات* النشطة. إذا قمت بتشغيل **systemctl status**، يعرض الإخراج نظرة عامة هرمية للخدمات قيد التشغيل. بمقارنة كل من المخرجات، ترى على الفور أن هناك أنواعًا متعددة من الوحدات وأن الخدمات واحدة فقط بينها.

يتم تمثيل كل خدمة *بوحدة خدمة -service unit-*، والتي يتم وصفها بملف خدمة يتم شحنها عادةً في **/lib/systemd/system/** (أو **/run/systemd/system/**، أو **/etc/systemd/system/** ؛ يتم إدراجها عن طريق زيادة ترتيب الأهمية، وآخر واحد يفوز). ربما يتم تعديل كل منها عن طريق ملفات *service-name*.**service.d/*.conf** أخرى في نفس مجموعة المجلدات. ملفات الوحدات هذه هي ملفات نصية عادية تعرف بامتداد "*.ini" أحيانا المعروفة في Microsoft Windows، مع أزواج *key = value* مجمعة بين رؤوس [*section*]. نرى هنا ملف خادم بسيط لـ **/lib/systemd/system/ssh.service**:

```
[Unit]
Description=OpenBSD Secure Shell server
After=network.target auditd.service
ConditionPathExists=!/etc/ssh/sshd_not_to_be_run
```

```
[Service]
EnvironmentFile=-/etc/default/ssh
ExecStart=/usr/sbin/sshd -D $SSHD_OPTS
ExecReload=/bin/kill -HUP $MAINPID
KillMode=process
Restart=on-failure
RestartPreventExitStatus=255
Type=notify

[Install]
WantedBy=multi-user.target
Alias=sshd.service
```

الوحدات المستهدفة هي جزء آخر من تصميم النظام. تمثل الحالة المرغوبة التي تريد تحقيقها من حيث الوحدات النشطة (مما يعني خدمة جارية في حالة وحدات الخدمة). وهي موجودة بشكل أساسي كوسيلة لتجميع التبعيات على الوحدات الأخرى. عندما يبدأ النظام، فإنه يمكّن الوحدات المطلوبة للوصول إلى **default.target** (وهي وصلة رمزية لـ **graphical.target** والذي يعتمد بدوره على **multi-user.target**). لذلك يتم تنشيط جميع تبعيات تلك الأهداف أثناء الإقلاع.

يتم التعبير عن هذه التبعيات بتوجيه **Wants** على الوحدة المستهدفة. ولكن ليس عليك تعديل الوحدة المستهدفة لإضافة تبعيات جديدة، يمكنك أيضًا إنشاء وصلة رمزية تشير للوحدة التابعة في المجلد **/etc/systemd/system/***target-name***.target.wants/**. وهذا بالضبط ما يفعله **systemctl enable** *foo.service*. عندما تقوم بتمكين خدمة، فأنت تخبر systemd أن يضيف تبعية على الأهداف المدرجة في إدخال **WantedBy**

لقسم [**install**] لملف وحدة الخدمة. عكس ذلك، يقوم **systemctl** **disable** *foo.service* بتعطيل نفس الوصلة الرمزية وبالتالي التبعية.

أمرا **enable** و **disable** لا تغير أي شيء يتعلق بالحالة الحالية للخدمات. إنهم تؤثران فقط على ما سيحدث في الإقلاع التالي. إذا كنت ترغب في تشغيل الخدمة على الفور، فيجب عليك تشغيل: **systemctl start** *foo.service*. على العكس من ذلك، يمكنك إيقافه من خلال **systemctl stop** *foo.service*. يمكنك أيضًا فحص الحالة الحالية للخدمة باستخدام: **systemctl status** *foo.service*، والتي تتضمن بشكل مفيد أحدث أسطر من السجل المرتبط. بعد تغيير تكوين الخدمة، قد ترغب في إعادة تحميلها أو إعادة تشغيلها: تتم هذه العمليات باستخدام: **systemctl reload** *foo.service* و **systemctl restart** *foo.service* على التوالي.

```
# systemctl status postgresql
● postgresql.service - PostgreSQL RDBMS
  Loaded: loaded (/lib/systemd/system/postgresql.service; disabled; vendor preset: disabled)
  Active: inactive (dead)

# ls -al /etc/systemd/system/multi-user.target.wants/postgresql.service
ls: cannot access '/etc/systemd/system/multi-user.target.wants/postgresql.service': No such file or directory
```

systemctl enable postgresql

[...]

ls -al /etc/systemd/system/multi-user.target.wants/postgresql.service

lrwxrwxrwx 1 root root 38 Apr 21 16:21 /etc/systemd/system/multi-user.target.wants/postgresql.service -> /lib/systemd/system/postgresql.service

systemctl status postgresql

● postgresql.service - PostgreSQL RDBMS

Loaded: loaded (/lib/systemd/system/postgresql.service; enabled; vendor preset: disabled)

Active: inactive (dead)

systemctl start postgresql

systemctl status postgresql

● postgresql.service - PostgreSQL RDBMS

Loaded: loaded (/lib/systemd/system/postgresql.service; enabled; vendor preset: disabled)

Active: active (exited) since Thu 2016-04-21 16:22:29 EDT; 2s ago

Process: 6355 ExecStart=/bin/true (code=exited, status=0/SUCCESS)

Main PID: 6355 (code=exited, status=0/SUCCESS)

Apr 21 16:22:29 kali-rolling systemd[1]: Starting PostgreSQL RDBMS...

Apr 21 16:22:29 kali-rolling systemd[1]: Started PostgreSQL RDBMS.

٥.٥. الملخص

تعلمنا في هذا الفصل كيفية تكوين Kali Linux. قمنا بتكوين إعدادات الشبكة، وتحدثنا عن المستخدمين والمجموعات، وناقشنا كيفية إنشاء وتعديل حسابات المستخدمين، وتعيين كلمات المرور، وتعطيل الحسابات، وإدارة المجموعات. أخيرًا، ناقشنا الخدمات وشرحنا كيفية إعداد الخدمات العامة وصيانتها، وتحديدًا SSH و PostgreSQL و Apache.

نصائح الملخص:

❖ في التثبيت النموذجي لسطح المكتب، سيكون لديك NetworkManager مثبتًا بالفعل ويمكن التحكم فيه وتكوينه من خلال مركز التحكم في GNOME ومن خلال القائمة العلوية اليمنى.

❖ يمكنك تكوين الشبكة من خلال سطر الأوامر باستخدام أدوات **ifup** و **ifdown**، التي تقرأ تعليماتها من ملف التكوين **/etc/network/interfaces**. أداة أحدث، *systemd-networkd* تعمل مع نظام *systemd* init.

❖ بشكل افتراضي، تتكون قاعدة بيانات مستخدمي ومجموعات Unix من ملفات نصية **/etc/passwd** (قائمة المستخدمين)، **/etc/shadow** (كلمات المرور المشفرة للمستخدمين)، **/etc/group** (قائمة المجموعات)، و **/etc/gshadow** (كلمات المرور المشفرة للمجموعات).

❖ يمكنك استخدام الأمر **getent** لاستشارة قاعدة بيانات المستخدم وقواعد بيانات النظام الأخرى.

❖ يطرح أمر **adduser** بعض الأسئلة قبل إنشاء الحساب، ولكنها الطريقة المباشرة لإنشاء حساب مستخدم جديد.

❖ يمكن استخدام عدة أوامر لتعديل حقول معينة في قاعدة بيانات المستخدم بما في ذلك: **passwd** (تغيير كلمة المرور)، **chfn** (تغيير الاسم الكامل و**GECOS**، أو حقل المعلومات العامة)، **chsh** (تغيير تسجيل الدخول الصدفة)، **chage** (تغيير عمر كلمة المرور)،و **passwd -e** *user* (يجبر المستخدم على تغيير كلمة المرور الخاصة به في المرة التالية التي يقوم فيها بتسجيل الدخول).

❖ يمكن لكل مستخدم أن يكون عضوا في مجموعة واحدة أو مجموعات متعددة. يمكن استخدام عدة أوامر لتعديل هوية المجموعة: يغير **newgrp** معرف المجموعة الحالي، **sg** ينفذ أمرًا باستخدام المجموعة البديلة المزودة، ويمكن وضع بت **setgid** في مجلد، مما يؤدي إلى أن تنتمي الملفات التي تم إنشاؤها في هذا المجلد تلقائيًا إلى المجموعة الصحيحة. بالإضافة إلى ذلك، يعرض الأمر **id** الحالة الحالية للمستخدم بما في ذلك قائمة بعضوية مجموعته.

❖ يمكنك بدء SSH يدويًا باستخدام **systemctl start ssh** أو تمكينه بشكل دائم باستخدام **systemctl enable ssh**. يعطل التكوين الافتراضي عمليات تسجيل الدخول المستندة إلى كلمة المرور للمستخدم الجذر، مما يعني أنه يجب عليك أولاً إعداد مفاتيح SSH باستخدام **ssh-keygen**.

- ❖ PostgreSQL هو خادم قاعدة بيانات. نادرًا ما يكون مفيدًا من تلقاء نفسه ولكن يتم استخدامه من قبل العديد من الخدمات الأخرى لتخزين البيانات.

- ❖ يشتمل التثبيت النموذجي لـ Kali Linux على خادم الويب Apache، الذي توفره حزمة apache2. كونها خدمة شبكة، يتم تعطيلها بشكل افتراضي. يمكنك تشغيله يدويًا باستخدام **systemctl start apache2**.

- ❖ من خلال التكوين الافتراضي، يستمع Apache على المنفذ 80 (كما تم تكوينه في **/etc/apache2/ports.conf**)، ويقدم صفحات من المجلد **/var/www/html/** افتراضيًا (كما تم تكوينه في **/etc/apache2/sites-enabled/000-default.conf**).

الآن بعد أن تعاملنا مع أساسيات Linux وتثبيت Kali Linux وتكوينه، دعنا نناقش كيفية تحرّي الخلل وإصلاحه وتعليمك بعض الأدوات والحيل لإعادتك للعمل عند مواجهة المشاكل.

التمرين الأول، الفصل الخامس - تكوين المستخدمين

١. قم بإنشاء حساب مستخدم قياسي. أضف المستخدم الجديد إلى مجموعة "sudo"

الإجابة:

```
adduser username
passwd username
usermod -a -G sudo username
chsh -s /bin/bash username
```

التمرين الثاني، للفصل الخامس - تكوين الشبكة

٢. أوقف خدمة Network Manager وقم بتعطيلها بالكامل في وقت الإقلاع.

٣. تكوين جهاز Kali الخاص بك لـ DHCP على eth0

٤. إنزال واجهة eth0.

٥. اتصل بالشبكة اللاسلكية باستخدام دونجل USB اللاسلكي الخاص بك عن طريق تكوين /etc/network/interfaces وفقًا لذلك.

الإجابة:

١. مدير الشبكة مفيد، ولكن في اختبار الإختراق تحتاج حقًا إلى الاستيلاء على واجهاتك وثنيها حسب إرادتك دون أي مفاجآت. لإيقاف Network Manager وتعطيله في وقت الإقلاع:

```
systemctl stop NetworkManager.service
systemctl disable NetworkManager.service
```

يمكنك التحقق من حالة الواجهات المُدارة في Network Manager من خلال:

```
nmcli dev status
```

نصيحة احترافية: أوقف مدير الشبكة من خلال إضافة dns-servers إلى ملف /etc/resolv.conf:

```
nano /etc/NetworkManager/NetworkManager.conf
```

أضف dns=none لقسم [main].

٢. اضبط eth0 لـ DHCP. قم بتغير ملف /etc/network/interfaces لتضمين:

```
auto eth0
iface eth0 inet dhcp
```

يمكنك أيضًا إعداد عنوان ثابت باستخدام ما يلي:

```
auto eth0

iface eth0 inet static

       address 192.168.1.160

       netmask 255.255.255.0

       gateway 192.168.1.1
```

٣. إنزال واجهة eth0:

```
ifconfig eth0 down
```

٤. الاتصال بشبكة لاسلكية. لاحظ أنه إذا كنت في جهاز إفتراضي، فستحتاج إلى محول لاسلكي USB. يفترض هذا المثال WPA2.Generate psk باستخدام الأمر التالي:

```
wpa_passphrase myssid wpa-password
```

الآن قم بإدراج PSK مع ما يلي داخل ملف /etc/network/interfaces:

```
auto wlan0

iface wlan0 inet dhcp

    wpa-ssid myssid

    wpa-psk {whatever the psk hash was}
```

قم بتدوير الواجهة:

```
ifup wlan0
```

التمرين الثالث، للفصل الخامس - تكوين الخدمات الجزء ١

١. تكوين SSH للسماح بتسجيل الدخول الجذر باستخدام كلمة المرور (تلميح: PermitrootLogin).
٢. ابدأ تشغيل خدمة SSH واتصل بها من النظام المضيف كمستخدم root.
٣. تكوين خدمة SSH للبدء في وقت الإقلاع.
٤. قم بتغيير كلمة مرور الجذر وقم بإنشاء مفاتيح مضيف SSH جديدة.
٥. النينجا! اجعل نسخة Kali الخاصة بك نقطة وصول عن طريق تثبيت **hostapd** وبدء تشغيله في وقت الإقلاع. قم بذلك بتكوين خدمة نظام مخصص! *هذا الجزء من التمرين قيد الاختبار.*

الإجابة:

١. عين **PermitrootLogin** لـ **yes** في /etc/ssh/sshd_config/

٢. ابدء sshd:

systemctl start ssh

٣. تمكين sshd عند الإقلاع:

systemctl enable ssh

٤. لأسباب أمنية، قم بتغيير كلمة مرور الجذر وقم بإنشاء مفاتيح مضيف SSH جديدة:

```
root@kali:~# passwd
[...]
root@kali:~# rm /etc/ssh/ssh_host_*
root@kali:~# dpkg-reconfigure openssh-server
root@kali:~# service ssh restart
```

٥. النينجا فقط! hostapd (برنامج نقطة الوصول للمضيف) هو نقطة وصول لبرامج مساحة المستخدم قادرة على تحويل بطاقات واجهة الشبكة العادية إلى نقاط وصول وخوادم مصادقة. لتهيئة خدمة hostapd يدويًا من خلال systemd:

تثبيت وتكوين المتطلبات الأساسية:

```
apt-get install hostapd
nano /etc/systemd/system/hostapd.service
```

أضف الاختبار التالي لملف hostapd.service:

```
[Unit]
Description=Hostapd WPE Service
After=network.target

[Service]
Type=simple
User=root
ExecStart=/usr/sbin/hostapd /etc/hostapd/hostapd.conf
Restart=on-abort

[Install]
WantedBy=multi-user.target
```

- قم بإنشاء أو نسخ ملف hostapd.conf إلى /etc/hostapd/hostapd.conf
- تعطيل مدير الشبكة! أعد تشغيل الخدمة وتمكينها في وقت الإقلاع. تأكد من أن hostapd يعمل بالفعل عند بدء الخدمة.

```
systemctl stop NetworkManager.service
systemctl disable NetworkManager.service
sudo nmcli radio wifi off
sudo rfkill unblock wlan
systemctl enable hostapd
systemctl start hostapd
ps -ef |grep hostapd
systemctl status hostapd
systemctl stop hostapd
ps -ef |grep hostapd
```

ملاحظة: إذا كنت تعمل على جهاز افتراضي، أو كنت تستخدم بطاقة Atheros، فقد تواجه مشكلات ("EEPROM magic" أو فشل البرامج الثابتة، وما إلى ذلك) مع المحول اللاسلكي المستند إلى USB. إذا كانت هذه هي الحالة، أخرج "eject" المحول في إعدادات VM الخاصة بك، افصله، أغلق VM بشكل سليم. أدخل البطاقة وقم بتشغيل الجهاز الافتراضي. إذا لم ينجح أي من هذا، فلا تقلق. هذا أمر صعب للغاية خاصة بسبب VM.

ملاحظة: **systemctl status hostapd** هو pal استكشاف الأخطاء وإصلاحها.

التمرين الرابع، الفصل الخامس – تكوين الخدمات الجزء الثاني

في هذا التمرين، سنقوم بتثبيت masscan. هذه أداة رائعة وسيساعد التثبيت الكامل في مراجعة بعض مفاهيم التكوين التي استكشفناها في هذا الفصل. يتم تقسيم العملية إلى عدة خطوات:

١. قم بتثبيت واباتشي خدمات PostgreSQL.
٢. كوِّن أباتشي و PostgreSQL للبدء في وقت الإقلاع.
٣. قم بتثبيت masscan، وهي متطلبات مسبقة وواجهة ويب ماسكان للأمن الشامل. استخدم حزم Apache / PostgreSQL.

Install masscan, it's prerequisites and Offensive Security's masscan web interface. Use an Apache / PostgreSQL stack.

٤. استيراد فحص سابق واعرض النتائج.
٥. قم بحماية تثبيت Apache باستخدام htaccess اسم مستخدم / كلمة مرور.

الإجابات:

سيكون هذا الحل معطلاً قليلاً. للبدء، راجع نسخة من مستودع masscan-web-ui:

```
root@kali:~# cd /root/
```

```
root@kali:~# git clone https://github.com/offensive-security/masscan-web-ui
```

بعد ذلك، تأكد من وجود جميع متطلبات masscan الرئيسية ونسخها عبر ملفات واجهة الويب MASSCAN إلى جذر الويب. لاحظ أنه إذا كنت تقوم بنسخ ولصق سطر **apt-get**، فهذا طويل. تأكد من انتزاع كل شيء:

```
root@kali:~# apt-get install apache2 php libapache2-mod-php php-xml postgresql php-pgsql
mv masscan-web-ui/* /var/www/html/
rm /var/www/html/index.html
```

إبدء تشغيل Apache و Postgres:

```
systemctl start apache2
systemctl start postgresql
```

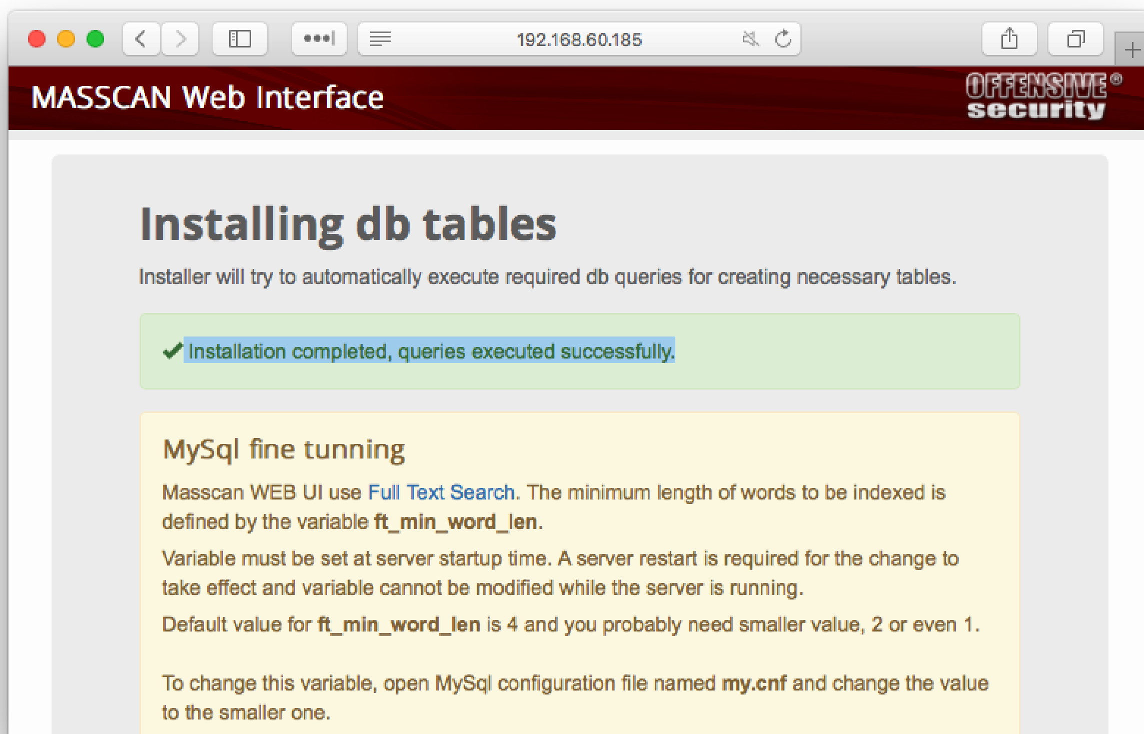
مع بدء Apache و PostgreSQL، تثبيت masscan و index.html الافتراضي من Apache، يمكنك تصفح خادم Apache الخاص بك لرؤية ماسكان. ومع ذلك، فإنه يشكو بحق من فشل مصادقة كلمة المرور. دعنا نصلح ذلك. إنشاء مستخدم ماسكان وإنشاء قاعدة بيانات ماسكان.

```
root@kali:~# su - postgres
postgres@kali:~$ createuser -P masscan
Enter password for new role:
Enter it again:
postgres@kali:~$ createdb -T template0 -E UTF-8 -O
masscan masscandb
exit
```

بعد ذلك، قم بتعديل كلمة المرور التي قمت بتعيينها واسم قاعدة البيانات (masscandb) في أسطر **define**:

```
root@kali:~# nano /var/www/html/config.php
[...]
root@kali:~# grep ^define /var/www/html/config.php
define('DB_DRIVER',         'pgsql');
define('DB_HOST',        '127.0.0.1');
define('DB_USERNAME',  'masscan');
define('DB_PASSWORD',  'toortoor');
define('DB_DATABASE',  'masscandb');
```

تصفح http://localhost على جهازك المحلي (أو العنوان البعيد إذا كنت تتصفح من خارج الجهاز الإفتراضي) والذي يجب أن يشير إلى أنه تم إعداد ماسكان بشكل صحيح:



بعد ذلك، دعنا نستورد بعض نتائج الفحص من فحص تم تشغيله مسبقًا. سنقوم بمسح قاعدة البيانات لأن هذه هي المرة الأولى التي نستخدم فيها ماسكان:

```
root@kali:~# wget https://kali.training/downloads/masscan.xml
root@kali:~# php /var/www/html/import.php /root/masscan.xml
Do you want to clear the database before importing (yes/no)?: yes

Clearing the db
Reading file
Parsing file
Processing data (This may take some time depending on file size)
```

```
Summary:
Total records:4646
Inserted records:4646
Took about:10 seconds
root@kali:~#
```

بعد ذلك، تصفح http://localhost لعرض البيانات المستوردة. نظرًا لأن هذه "بيانات حساسة"، فإننا نريد حماية مجلد الويب الجذر بكلمة مرور. للقيام بذلك، يجب أن نبدأ بتوجيهات Apache:

```
root@kali:~# nano /etc/apache2/sites-enabled/000-default.conf
```

أضف هذه الأسطر:

```
AuthType Basic
AuthName "Restricted Content"
AuthUserFile /etc/apache2/htpasswd
Require valid-user
```

ودعنا ننشئ بيانات اعتماد لمستخدم جديد:

```
root@kali:~# htpasswd -c /etc/apache2/htpasswd myuser
```

أخيرًا، استعرض تصفح http://localhost، وأدخل بيانات اعتمادك واعرض التقرير.

هل تعلم؟

هل تتذكر سياسة Kali Linux لتعطيل خدمات الشبكة افتراضيًا؟ تم تكوين هذه السياسة من

/lib/systemd/system-preset/{95-kali.preset،99-default.preset}

نقطة وصول راسبيري باي

إذا لم يكن لديك Raspberry Pi 3، فعليك الحصول على واحدة. فهي رائعة للغاية وغير مكلفة نسبياً. في هذا التمرين، ستقوم بتكوين Raspberry Pi 3 ليتم تشغيله كنقطة وصول لاسلكية، مما يمنح المستخدمين المتصلين إمكانية الوصول إلى الإنترنت. هذا التمرين رائع لأنك ستقوم بتثبيت Kali على Raspberry Pi وتعديل الملفات وتغيير أذونات الملفات وتكوين واجهات الشبكة وتثبيت الخدمات وتكوينها وتكوين قواعد iptables والمزيد. إنها نظرة عامة رائعة.

إليك ما عليك القيام به:

١. قم بتثبيت Kali على Raspberry Pi 3. يمكنك استخدام صورة مخصصة، ولكن إذا قمت بذلك، فقد يكون لديك المزيد من استكشاف الأخطاء وإصلاحها. إذا لم تكن متأكدًا، فاستخدم صورة المخزون التي تمت كتابة هذا الحل من أجلها.

٢. تطبيق أمان WPA2 على AP.

٣. قم بتكوين **eth0** على أنه DHCP، و**wlan0** على أنه ثابت.

٤. قم بتكوين Raspberry Pi كخادم DHCP لأي عميل لاسلكي وتعيين مصادقة بـ DHCP لمدة ١٢ ساعة.

٥. اجعل خادم SSH يبدأ في وقت الإقلاع حتى تمكن SSH إلى Raspberry Pi بمجرد تشغيله.

٦. إعادة توجيه كل حركة المرور الصادرة، بما في ذلك DNS، من **wlan0** إلى **eth0**.

٧. السماح بالاتصالات الداخلية (ذات الحالة) الواردة من **eth0** إلى **wlan0**.

٨. تلميح: على الرغم من أنك لم تتعلم عن **hostapd** أو **dnsmasq**، إلا أنك ستستخدمها في هذا التمرين.

٩. الغش الجزئي: على الرغم من أن هذا المقال لم يكتب لكالي (ولن يعمل كما هو مكتوب في كالي)، إلا أنه مصدر إلهام لهذا التمرين، ويستحق المراجعة. بفضل فيل مارتن للإلهام.

الإجابات:

سيتطلب هذا بعض الأشياء:

- ❖ Raspberry Pi 3: يمكنك استخدام طراز أقدم بشبكة wifi USB ولكنك لوحدك عندما يتعلق الأمر بتكوين **wlan0**.
- ❖ hostapd: يؤدي هذا إلى إنشاء نقطة اتصال.
- ❖ dnsmasq: يقوم هذا بإعادة توجيه DNS ويوفر قطع DHCP.
- ❖ dhcpcd5: عميل DHCP (الذي يقوم أيضًا بأشياء أخرى رائعة لإدارة الشبكة).

احصل على الحزم المطلوبة:

```
apt-get install dnsmasq hostapd dhcpcd5
```

أولاً، دعنا نطلب من dhcpcd تجاهل إعداد wlan0. سنقوم بتكوين عنوان IP ثابت لاحقًا:

```
nano /etc/dhcpcd.conf
```

ضع هذا فوق أي سطور واجهة قد تكون في الملف:

```
denyinterfaces wlan0
```

الآن، فلنقم بإعداد واجهة wifi الخاصة بنا. إذا كان لديك Pi 2 مع محول USB wi-fi، فتابع وقم بتوصيله الآن. تحرير ملف الواجهات:

```
nano /etc/network/interfaces
```

وأضف هذا القسم:

```
allow-hotplug wlan0

iface wlan0 inet static

    address 172.24.1.1

    netmask 255.255.255.0
```

```
network 172.24.1.0
broadcast 172.24.1.255
```

أعد تشغيل dhcpcd باستخدام:

```
root@kali:~# service dhcpcd restart
```

ثم أعد تحميل تكوين wlan0 باستخدام:

```
root@kali:~# ifdown wlan0; ifup wlan0
```

بعد ذلك، فلنقم بتكوين hostapd بملف تكوين جديد. لاحظ أنه تم تكوين SSID وكلمة المرور لنقطة الوصول الخاصة بك.

```
root@kali:~# nano /etc/hostapd/hostapd.conf
[..]
root@kali:~# cat /etc/hostapd/hostapd.conf

# This is the name of the WiFi interface we configured above
interface=wlan0

# Use the nl80211 driver with the brcmfmac driver
driver=nl80211

# This is the name of the network
ssid=Kali-Pi3
```

```
# Use the 2.4GHz band
hw_mode=g

# Use channel 6
channel=6

# Enable 802.11n
ieee80211n=1

# Enable WMM
wmm_enabled=1

# Enable 40MHz channels with 20ns guard interval
ht_capab=[HT40][SHORT-GI-20][DSSS_CCK-40]

# Accept all MAC addresses
macaddr_acl=0

# Use WPA authentication
auth_algs=1

# Require clients to know the network name
ignore_broadcast_ssid=0

# Use WPA2
wpa=2

# Use a pre-shared key
```

```
wpa_key_mgmt=WPA-PSK

# The network passphrase
wpa_passphrase=raspberrytoor

# Use AES, instead of TKIP
rsn_pairwise=CCMP
```

عند هذه النقطة، يمكننا اختبار الأشياء. شغل:

```
root@kali:~# /usr/sbin/hostapd /etc/hostapd/hostapd.conf
```

هذا يدل على تشغيل ناجح. لاحظ أن الأخطاء المتعلقة بوضع المراقبة ليست ذات صلة بنا. بالنسبة إلى RPi3 باستخدام برنامج nexmon، نحتاج (أو تطبيق) الـ *nexutil -m2*.

```
root@kali:~# /usr/sbin/hostapd /etc/hostapd/hostapd.conf
Configuration file: /etc/hostapd/hostapd.conf
Failed to create interface mon.wlan0: -95 (Operation not supported)
wlan0: Could not connect to kernel driver
Using interface wlan0 with hwaddr b6:ae:d7:42:a1:70 and ssid "Kali-Pi3"
wlan0: interface state UNINITIALIZED->ENABLED
wlan0: AP-ENABLED
```

يمكنك الاتصال بنقطة الوصول هذه وسيعرض hostapd بعض الإخراج:

wlan0: STA 78:4f:43:7c:6d:32 IEEE 802.11: associated

وبمجرد إدخال كلمة المرور، سترى شيئًا مثل هذا:

wlan0: AP-STA-CONNECTED 78:4f:43:7c:6d:32

wlan0: STA 78:4f:43:7c:6d:32 RADIUS: starting accounting session 5991CC2F-00000000

wlan0: STA 78:4f:43:7c:6d:32 WPA: pairwise key handshake completed (RSN)

wlan0: STA 78:4f:43:7c:6d:32 IEEE 802.11: disassociated

wlan0: AP-STA-DISCONNECTED 78:4f:43:7c:6d:32

wlan0: INTERFACE-DISABLED

wlan0: STA 00:00:00:00:00:00 IEEE 802.11: disassociated

wlan0: INTERFACE-ENABLED

wlan0: STA 78:4f:43:7c:6d:32 IEEE 802.11: associated

لاحظ أن عميلك ربما ينقطع الاتصال به ويعاد الاتصال لأنه لم يحصل على عنوان IP. هذا امر طبيعي. لن تحصل على عنوان IP حتى نقوم بتكوين dnsmasq. استمتع بهذا! يمنحك فكرة عن كيفية عمل هذه العملية، خلف الكواليس.

اضغط على Ctrl-C لإيقاف **hostapd**.

بعد ذلك، سنخبر hostapd بمكان العثور على ملف التكوين الخاص به:

```
root@kali:~# nano /etc/default/hostapd
```

ابحث عن سطر ""=DAEMON_CONF# واستبدله بـ:

DAEMON_CONF="/etc/hostapd/hostapd.conf"

لنقم بتعديل **dnsmasq**:

```
root@kali:~# mv /etc/dnsmasq.conf /etc/dnsmasq.conf.orig
root@kali:~# nano /etc/dnsmasq.conf
```

يجب أن يبدو كالتالي:

```
interface=wlan0        # Use interface wlan0
listen-address=172.24.1.1 # Set our listening address
bind-interfaces        # Bind to the interface to make sure we aren't sending things elsewhere
server=8.8.8.8         # Forward DNS requests to Google DNS
domain-needed          # Don't forward short names
bogus-priv             # Never forward addresses in the non-routed address spaces.
dhcp-range=172.24.1.50,172.24.1.150,12h   # Assign IP addresses between 172.24.1.50 and 172.24.1.150 with a 12 hour lease time
```

الأن لدينا واجهتان نشطتان، وعندنا عميل DHCP ل الروسفيري الخاص بنا وخادم DHCP لمضيفي الوايرليس الخاص بنا. الأن نحتاج لإعادة توجيه حركة المرور بين واجهات wifi و ethernet. يمكننا تحقيق ذلك على الفور باستخدام أمر بسيط لتحديث proc/:

```
root@kali:~# echo 1 > /proc/sys/net/ipv4/ip_forward
root@kali:~# cat /proc/sys/net/ipv4/ip_forward
1
```

ومع ذلك، لن يلتزم هذا التغيير بين عمليات إعادة التشغيل. نحن بحاجة إلى جعلها دائمة من خلال sysctl:

```
root@kali:~# nano /etc/sysctl.conf
```

إلغي تعليق السطر الذي يحتوي على **net.ipv4.ip_forward =1**:

```
root@kali:/var/www/html# grep ip_forward /etc/sysctl.conf
net.ipv4.ip_forward = 1
```

إن عملية إعادة التوجيه ليست كافية تمامًا لمنح مضيفي wifi الخاصين بنا إمكانية الوصول إلى الإنترنت (من خلال واجهة eth0). نحن بحاجة إلى iptables لمساعدتنا على القيام بذلك.

```
root@kali:~# iptables -t nat -A POSTROUTING -o eth0 -j MASQUERADE
root@kali:~# iptables -A FORWARD -i eth0 -o wlan0 \
> -m state --state RELATED,ESTABLISHED -j ACCEPT
root@kali:~# iptables -A FORWARD -i wlan0 -o eth0 -j ACCEPT
```

دعونا نبسط هذه الأوامر:

١. عندما يتم العثور على اتصال جديد (`-t nat`)، نريد تبديل -alter- الحزم لأنها على وشك الخروج (-A POSTROUTING) على واجهة إيثرنت (-o eth0). الهدف -j MASQUERADE يحجب عنوان IP الخاص للعميل بعنوان IP الخارجي لجدار الحماية / البوابة (Kali Pi).

٢. بعد ذلك، نلحق (-A) بقاعدة إلى سلسلة FORWARD (يتم توجيه الحزم عبر Pi) والتي تقبل (-j ACCEPT) الحزم من eth0 إلى wlan0 (-i eth0 –o wlan0) التي تنتمي إلى (ESTABLISHED) أو المتعلقة (RELATED) باتصال موجود.

٣. أخيرًا، سنعيد توجيه -forward - (ونقبل -accept-) جميع الحزم من wlan0 إلى eth0.

تحقق من قواعدنا:

```
root@kali:~# iptables -S
-P INPUT ACCEPT
-P FORWARD ACCEPT
-P OUTPUT ACCEPT
-A FORWARD -i eth0 -o wlan0 -m state --state RELATED,ESTABLISHED -j ACCEPT
-A FORWARD -i wlan0 -o eth0 -j ACCEPT
```

إخراج قواعدنا إلى ملف:

```
iptables-save > /etc/iptables.ipv4.nat
```

قم بتطبيق هذه القواعد في كل مرة نقوم فيها بتشغيل Pi عن طريق تحرير ملف **/etc/rc.local**:

```
root@kali:~# nano /etc/rc.local
[...]
root@kali:~# more /etc/rc.local
#!/bin/sh -e
iptables-restore < /etc/iptables.ipv4.nat
```

اجعل الملف قابلاً للتنفيذ:

```
root@kali:~# chmod 711 /etc/rc.local
root@kali:~# ls -l /etc/rc.local
-rwx--x--x 1 root root 57 Aug 10 19:37 /etc/rc.local
```

كما رأينا، يتم شحن **hostapd** و **dnsmasq** مع جميع مزايا نظام التهيئة -*init system*- (انظر /etc/init.d)، لذلك دعونا نبدأ الخدمات ونتحقق منها:

```
root@kali:~# systemctl start hostapd dnsmasq
root@kali:~# systemctl status hostapd dnsmasq
● hostapd.service - LSB: Advanced IEEE 802.11 management daemon
   Loaded: loaded (/etc/init.d/hostapd; generated; vendor preset: disabled)
   Active: active (running) since Mon 2017-08-14 19:24:43 UTC; 2s ago
[...]
● dnsmasq.service - dnsmasq - A lightweight DHCP and caching DNS server
```

```
   Loaded:   loaded   (/lib/systemd/system/dnsmasq.service;
disabled; vendor preset:

   Active: active (running) since Mon 2017-08-14 19:24:43
UTC; 2s ago
```

ولتمكنها للعمل بعد إعادة الإقلاع:

```
root@kali:~# systemctl enable hostapd dnsmasq
hostapd.service is not a native service, redirecting to systemd-sysv-
install.
Executing: /lib/systemd/systemd-sysv-install enable hostapd
Synchronizing state of dnsmasq.service with SysV service script with
/lib/systemd/systemd-sysv-install.
Executing: /lib/systemd/systemd-sysv-install enable dnsmasq
```

أخيرًا، أعد التشغيل وتأكد من الالتزام بالقواعد بعد إعادة التشغيل. بمجرد إعادة التشغيل، يجب أن تكون قادرًا على الاتصال بـ "Kali Pi" والتصفح!

تمرين الشهادة للفصل الخامس

١. بأي أداة يمكنك التحكم في الشبكة في الواجهة الرسومية لـ gnome؟

- ifupdown
- systemctl
- NetworkManager
- /etc/network/interfaces

٢. يعد ملف الواجهات جزءًا مهمًا من تكوين الشبكة بسطر الأوامر. ما هو المجلد الخاص بها؟

- /etc/networks
- /etc/init.d
- /etc/network
- /etc/init

٣. ما هو اسم حزمة سطر الأوامر المستخدمة عادة في كالي لتكوين الشبكة من سطر الأوامر؟

- systemctl
- init.d
- ifupdown
- hosts

٤. عند تكوين شبكة من سطر الأوامر (على سبيل المثال مع ifup أو ifdown) أي سطر سيبدأ القسم لتكوين شبكة يدوي؟

```
o iface eth0 inet auto
o iface eth0 inet auto
o iface eth0 inet auto
o iface eth0 inet static
```

٥. ما هي الأساليب التي يمكن استخدامها لتكوين أجهزة الشبكة في Kali Linux؟ اختر كل ما يمكن تطبيقه:

- o رسوميا باستخدام NetworkManager
- o بسطر الأوامر باستخدام ملفات network. في المجلد /etc/system/network
- o بسطر الأوامر بواسطة ملف /etc/network/interfaces
- o بسطر الأوامر بواسطة systemd-networkd
- o بسطر الأوامر باستخدام ifupdown

٦. أي ملف يحتوي على كلمات مرور المستخدم المشفرة؟

- o /etc/group
- o /etc/shadow
- o /etc/passwd
- o لاشيء مما سبق

٧. ما هو الأمر المستخدم لإضافة مستخدمين؟

- o passwd -l
- o adduser
- o chuser
- o useradd

٨. ما هو الأمر الذي سيعلق حساب المستخدم؟

- o **useradd -s** *olduser*
- o **passwd -l** *olduser*
- o **passwd -s** *olduser*
- o **rmuser -l** *olduser*

٩. ما هو الصحيح لخدمة SSH على تثبيت كالي الافتراضي؟ اختر كل ما ينطبق.

- ☐ يتم إنشاء المفاتيح الإفتراضية من صورة مباشرة مسبقا
- ☐ يحظر التكوين الافتراضي عمليات تسجيل الدخول المستندة إلى كلمة المرور
- ☐ يحظر ملف التكوين الافتراضي عمليات تسجيل الدخول المستندة إلى الشهادات
- ☐ تم تعطيل خدمة SSH بشكل افتراضي
- ☐ يتم تثبيت خدمة SSH بشكل افتراضي

١٠. ما هو الأمر الشائع استخدامه لبدء تشغيل خدمات مثل ssh و postgresql؟

- o service
- o systemctl
- o init
- o run

١١.	ما هو الأمر المستخدم لإضافة قاعدة بيانات postgresql جديدة؟

- o dropdb
- o createdb
- o psql -n
- o db_create

١٢.	أي من هذه الأوامر ليس من أوامر postgresql؟

- o createuser
- o pg_createuser
- o psql
- o createdb

١٣.	أي من هذه الأوامر تنشئ قاعدة بيانات postgres باسم db_new؟

- o psql -h localhost -c db_new -O dbuser dbuser
- o createdb -T template0 -E UTF-8 -O dbuser db_new
- o pg_create -o dbuser -n db_new -E UTF-8
- o createdb -T template0 -E UTF-8 -n db_new

١٤.	أي مما يلي ليس لها علاقة بـ Apache2؟ اختر واحدة.

- o a2enmod
- o systemctl start apache
- o /etc/apache2
- o /var/www/html

١٥.	أي مما يلي ليس له علاقة بـ Apache2؟

- /etc/apache2/mods-available
- /etc/apache2/ports.conf
- DocumentRoot
- htpasswd
- .htaccess
- Apachectl

١٦.	في كالي، ما هو المسؤول عن تسلسل الإقلاع، ولكنه يعمل أيضًا بصفة دائمة كمدير خدمة كامل الميزات وبدء الخدمات ومراقبتها؟

- init.d
- systemd
- grub
- systemctl

١٧.	أي أمر سيفحص الوضع الحالي لخدمة postgresql؟

```
o /etc/init/postgresql status
o ps | grep postgresql
o sudo status postgresql
o systemctl status postgresql
```

الإجابات الصحيحة:

1. NetworkManager
2. /etc/network
3. ifupdown
4. iface eth0 inet static
5. On the command line with .network files in the /etc/systemd/network directory

   On the command line with ifupdown

   On the command line via the /etc/network/interfaces file

   Graphically with NetworkManager

   On the command line with systemd-networkd
6. /etc/shadow
7. **adduser**
8. **passwd -l** *olduser*
9. The SSH service is installed by default

   The default configuration blocks password-based logins

   The SSH service is disabled by default

   The default keys from a live image are pre-generated
10. **systemctl**
11. **createdb**
12. pg_createuser
13. **createdb -T template0 -E UTF-8 -O dbuser db_new**
14. **systemctl start apache**
15. apachectl
16. systemd
17. **systemctl status postgresql**

أسئلة خطرت ببالي أثناء الدراسة

١- مالفرق بين systemctl و service؟

الجواب -مختصرا-: أن systemctl يعطي خيارات أكثر وتحكم أكبر.
أما service فالتحكم فيه للخوادم الأساسية فقط.

٢- كيف يمكنني تغيير المجموعة الحالية أو الأولية؟

```
#usermod -g (المجموعة الأولية الجديدة) (اسم المستخدم)
```

٣- كيف نزيل مستخدم من مجموعة؟

```
#deluser user group
```

أو:

```
#usermod -a -G group user
```